METHOD AND APPARATUS FOR CACHE SPACE ALLOCATION

Field of the Invention

The present invention relates generally to cache memory devices, and more particularly, to methods and apparatus for allocating a portion of a cache to a given task.

Background of the Invention

Processors often use a cache to improve performance and decrease system costs. Caches temporarily store recently accessed information (blocks of instructions or data) in a small memory that is faster to access than a larger main memory. Caches are effective because a block that has been accessed once is likely to be accessed soon again or is often near a recently accessed block. Thus, as a task executes, the working set of a task (the instructions and data currently required for the task) is stored in the cache in the event that the information may be accessed again. A cache typically maps multiple blocks of information from the main memory into one place in a cache, typically referred to as a "set." A "block" refers to the minimum unit of information that can be present in a cache and a "frame" is the place in a cache where a single block may be stored. In a set associative cache, multiple frames are grouped into sets. For example, as two-way set associative cache has two frames in each set.

In many embedded applications, a primary task may be interrupted by one or more secondary tasks. Thus, following an interruption, as a secondary, interrupting task executes, the working set of the interrupting task is stored in the cache, potentially evicting the working set of the primary, interrupted task and thereby decreasing the performance of the primary, interrupted task when it resumes execution. When the primary, interrupted task resumes execution, portions of the working set that have been evicted from the cache must be obtained from main memory, causing a "cache miss." Thus, the execution time of the primary, interrupted task is extended by the time taken to run the secondary task plus the miss penalty due to obtaining evicted portions of the cached information from the main memory.

A need therefore exists for a cache management technique that constrains one or more identified tasks to certain portions of a cache. In this manner, one or more secondary tasks may be allocated a certain section of the cache, preserving the unallocated section of the cache

for a primary task. In addition, a need exists for a cache management technique that allows a section of the cache to be allocated to one or more identified tasks.

Summary of the Invention

Generally, a method and apparatus are disclosed for allocating a section of a cache memory to one or more tasks. For example, one or more secondary tasks may be allocated a certain section of the cache, preserving the unallocated section of the cache for a primary task. The present invention transforms a set index value that identifies a corresponding set in the cache memory to a mapped set index value that constrains a given task to the corresponding allocated section of the cache. The allocated cache section of the cache can be varied by selecting an appropriate map function. When the map function is embodied as a logical and function, for example, individual sets can be included in an allocated section, for example, by setting a corresponding bit value to binary value of one.

A cache addressing scheme is disclosed that permits a desired portion of a cache to be selectively allocated to one or more tasks. In one implementation, a two-to-one multiplexer is employed for each set in the cache. A first register stores the desired size of a section and a second register stores a section selection. A size select value determines the size of an allocated segment and a section selection value determines the particular allocated segment that is selected. In this manner, a desired location and size of an allocated section of sets of the cache memory may be specified.

A more complete understanding of the present invention, as well as further features and advantages of the present invention, will be obtained by reference to the following detailed description and drawings.

Brief Description of the Drawings

FIG. 1 illustrates a cache allocation system in accordance with the present invention;

FIG. 2 illustrates a conventional scheme for addressing a cache, such as the cache of FIG. 1;

FIG. 3 illustrates a cache addressing scheme that permits portion of the cache to be allocated to one or more tasks in accordance with the present invention;

FIG. 4 illustrates a map function, M, that maps a set index, A, to a mapped set index, a, in accordance with the present invention; and

FIG. 5 illustrates a cache addressing scheme that permits a portion of the cache to be selectively allocated to one or more tasks in accordance with another embodiment of the present invention.

Detailed Description

FIG. 1 illustrates a cache allocation system 100 in accordance with the present invention. As shown in FIG. 1, the cache allocation system 100 allocates a section 140 of the cache 150 to one or more tasks. For example, one or more secondary tasks may be allocated a certain section of the cache. Thus, the secondary tasks may use only the allocated section of the cache, preserving the unallocated section for the primary task and consequently reducing the eviction of lives lines of the primary task. In this manner, the number of misses suffered on resumption of the primary task is reduced. It is recognized that limiting the cache space used by the secondary task may increase the misses for the secondary task. Thus, the benefits of the present invention are fully realized only in cases where the penalty on the secondary task is less than that experienced by the primary task due to eviction of lines accessed later. The benefit is most evident in the case where the secondary task is sequential and large relative to the cache. The cache allocation system 100 and cache 150 can be part of a digital signal processor (DSP), microcontroller, microprocessor, application specific integrated circuit (ASIC) or another integrated circuit.

FIG. 2 illustrates a conventional scheme for addressing a cache, such as the cache 150 of FIG. 1. As shown in FIG. 2, the exemplary cache 210 is a two-way set associative (two frames per set) 64 Kilobyte cache having 32-byte blocks. A portion of the 32 bit address of a block of main memory is a ten (10) bit set index value identifying the corresponding set in the cache 210. While the present invention may be incorporated into all cache organizations (data or instruction), the present invention is illustrated with a two-way set associative instruction cache that has two frames at each set address.